Distributed best response dynamics with high playing rates in potential games

Stéphane Durand, Federica Garin, Bruno Gaujal

Distributed Best Response dynamics with high playing rates in potential games

Stéphane Durand[a,b], Federica Garin[b], Bruno Gaujal[a,*]

[a] *Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, Grenoble, France*
[b] *Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab, Grenoble, France*

Abstract

In this paper we design and analyze distributed best response dynamics to compute Nash equilibria in potential games. This algorithm uses local Poisson clocks for each player, and does not rely on the usual but unrealistic assumption that players take no time to compute their best response. If this time (denoted δ) is taken into account, distributed best response dynamics may suffer from *overlaps*: one player starts to play while another player has not changed its strategy yet. Overlaps may lead to drops of the potential but we can show that they do not jeopardize eventual convergence to a Nash equilibrium. Our main result is to use a Markovian approach to show that the average execution time of the algorithm can be bounded from above by $e^{\gamma}\frac{\delta n \log n}{\log\log n}(1+o(1))$ and from below by $\frac{\delta n \log n}{\log\log n}(1+o(1))$, where γ is the Euler constant, n is the number of players and δ is the time taken by one player to compute its best response. These bounds are obtained by using an asymptotically optimal playing rate λ. Our analytic bounds show that this λ is high: $\hat{\lambda} = \frac{\log\log n - \log\log\log n}{\delta}$. This induces a large probability of overlap ($\hat{p} = 1 - \log\log n/\log n$). In practice, numerical simulations also show that using high playing rates is efficient, with an optimal probability of overlap $p_{opt} \approx 0.78$ up to $n = 250$. This implies that best response dynamics are unexpectedly efficient to compute Nash equilibria, even in a distributed setting.

Keywords: Potential Games, Best Response Dynamics, Distributed Algorithms, Markov Chains

1. Introduction

Potential games have first been introduced in [1]. They are both rampant in applications (routing and/or congestion games are potential games in general,

*Corresponding author
Email addresses: stephane.durand@inria.fr (Stéphane Durand), federica.garin@inria.fr (Federica Garin), bruno.gaujal@inria.fr (Bruno Gaujal)

as it was first noticed in [2]) and have been exhaustively studied *per se* (see [3]). They play a major role in transportation science as well as in computer science [4, 5, 6] and in distributed optimization [7].

It is well-known that the Best Response Dynamics (BRD) converges to a pure Nash equilibrium in potential games [3]. Here, we study the average execution time of the BRD and its dependence on the order of play of the players (called the *revision sequence* in the following), in a distributed context. When one uses BRD to compute a Nash equilibrium in potential games in practice, one is confronted with a mixed feeling.

On one hand, BRD with a round robin revision sequence has been proved optimal among all local search algorithms (converges faster than any local search in the strong stochastic sense in the set of all potential games with a uniform distribution), see [8].

On the other hand, BRD suffers from two main drawbacks when used in a distributed context. Firstly, the impact of the revision sequence on the performance is still unknown: If one replaces the round robin revision by another order of play, the convergence time will certainly grow, but the degradation still needs to be evaluated. This may be critical in cases where round robin among players is hard to implement, such as in distributed cases.

Secondly, convergence of BRD is only guaranteed when players play one at a time. Again, this constraint may hamper performance in a distributed context because electing a single active player at each round also requires costly synchronization between players.

In this paper we provide answers to both drawbacks at the same time. We study the case where the global order of plays results from local Poisson clocks for each player, all with the same rate, denoted λ/n. When all players play according to a local Poisson clock, then they play one at a time almost surely. This property is used in many distributed algorithms to avoid the difficult case of simultaneous actions. In practice, however, the time taken by one player to play cannot be neglected compared with the time gap between two plays. Similarly to what is called *collisions* in distributed communication protocols, when two players start to play "almost" simultaneously, their plays *overlap*: The second player starts playing before the first one has finished, resulting in an incoherent profile of the game.

In this paper we show that although overlaps will cause drops of the potential over time, they will not jeopardize convergence to a Nash equilibrium in finite time. We also show that the convergence time can be computed as a product of a function of the overlap probability and a function of the number of players and that a high probability of overlaps is even desirable to speed up convergence.

To achieve this, we introduce two successive simplifications whose behavior can be compared (using two different coupling schemes) with the best response dynamics over a potential game with n players. These simpler models are called respectively the Intersection-Free approximation (IFA) and the Restart approximation (RST), for reasons explained later. The convergence time of the simplest system (RST) can be evaluated by modeling its behavior by a Markov chain over a hybrid state space (with a continuous and a discrete component).

The expected hitting time of a Nash equilibrium for this Markov chain satisfies a Poisson equation. This Poisson equation can be transformed into an ordinary differential equation with an implicit initial condition whose solution can be computed in integral form. The derivation of the asymptotic behavior of this solution in n (the number of players) is the main technical part of this paper. It shows that the expected execution time for the restart approximation (and hence of BRD) is bounded by $\mathbb{E}[T_{BRD}] \leq e^{\gamma}\delta n \log(n)/\log\log n$ asymptotically, where δ is the time taken by a player to compute its best response and γ is the Euler constant, when the rate of play (called λ) is chosen appropriately.

This result calls for several comments. First, this is an illustration of the power of the approach used here: using a Markov model of the behavior of a distributed algorithm, based on its invariant state, is precise enough to obtain an upper bound on the time complexity that is comparable with a trivial lower bound of the complexity. Indeed, using the coupon collector problem, it is rather straightforward to see that $\mathbb{E}[T_{BRD}] \geq \frac{1}{\lambda} n H_n$, where $H_n = \sum_{j=1}^{n}(1/j)$ is the nth harmonic number.

Second, this shows that BRD qualifies as an efficient algorithm in practice to compute Nash equilibria in a centralized as well as in a distributed context, even when players do not share any global information and when overlaps occur. The average complexity lower than $n \log n$ is in sharp contrast with the worst case complexity of BRD (exponential in n, even for round robin centralized games).

To test the sensitivity of the execution time to the only free parameter (*i.e.* the rate of the Poisson clock λ), we have run several simulations of games of various sizes. Simulations show that the distribution of computing times are rather tight around the expected value. They also show some robustness of the convergence time with respect to the playing rate λ. This suggests several future improvements of the algorithm such as letting λ vary over time to adjust automatically to the number of players when it is unknown and to speed up convergence. Simulations also show that decreasing the chances of overlaps by reducing the play intensity λ is not a good idea (up to a certain point). Finally, while the analysis of our upper bound is minimized with a overlap probability $p \approx 1 - \log\log n/\log n$ (this is surprisingly high), the numerical experiments show that the best overlap probability is around $p = 0.78$, for any number of players, up to 250.

2. Random Potential Games

2.1. Potential Games and Best Response

A game $\mathfrak{G} \stackrel{\text{def}}{=} \mathfrak{G}(\mathcal{N}, \mathcal{A}, u)$ will be a triplet consisting of:

- A finite set of *players* $\mathcal{N} = \{1, \ldots, n\}$;
- A finite set $\mathcal{A}$ of *actions* (or *pure strategies*) ; The set of *actions profiles* or *states* of the game is $\mathcal{A}^n$;
- The players' *payoff functions* $u_k : \mathcal{A}^n \to \mathbb{R}$, for each $k \in \mathcal{N}$.

The number of actions per player is denoted A. We also use $a := A - 1$ the number of alternative actions per player in a given state.

The *best response correspondence* $\mathcal{BR}_k(x)$ is the set of actions maximizing the payoff for player k under state $\mathbf{x}$:

$$\mathcal{BR}_k(\mathbf{x}) \stackrel{\text{def}}{=} \left\{ \arg\max_{\alpha_k \in \mathcal{A}} u_k(\alpha_k; \mathbf{x}_{-k}) \right\}.$$

A *Nash equilibrium* (Nash equilibrium (NE)) is a fixed point of this correspondence, i.e., a profile x^* such that $x_k^* \in \mathcal{BR}_k(x^*)$ for every player k.

Definition 1 (Potential games). *A game is a* (best response) potential game *[9] if there exists a function* $\Phi : \mathcal{A}^n \to \mathbb{R}$ *such that for any player* k *and action profile* $\mathbf{x}$

$$\mathcal{BR}_k(\mathbf{x}) = \left\{ \arg\max_{\alpha_k \in \mathcal{A}} \Phi(\alpha_k, \mathbf{x}_{-k}) \right\}.$$

To avoid ties we assume that the Best Response is unique: $\mathcal{BR}_k(\mathbf{x}) = \arg\max_{\alpha_k \in \mathcal{A}} \Phi(\alpha_k; \mathbf{x}_{-k})$. Ties being of measure zero in the set of random games, they will not affect the average behavior of the players.

We consider an algorithmic version of the Best Response Dynamics driven by a revision sequence, called *Best Response Dynamics* (BRD) in the following.
100 A revision sequence is an infinite sequence of players chosen according to some rule. We will mostly consider two relatively natural sequences: the round robin sequence, i.e., the cyclic sequence $1, \cdots, n, 1, \cdots, n, 1, \cdots$, and an independent and uniformly distributed sequence resulting from the superposition of n independent Poisson point processes (one for each player), all with the same rate λ. Under this random revision sequence, each player is chosen according to a uniform law independently with probability $1/n$ at each point of the process. The latter will be useful to study the distributed version of the game.

Algorithm 1: Best Response Dynamics (BRD) under revision sequence R

```
1 Input: Game utilities (u_k(·)); Initial state (x := x(0)); revision
  sequence R; Initialize t := 0; List of satisfied customers L := ∅;
2 while size(L)≠ n do
3     Pick next player k := R_t; t := t + 1;
4     if x_k ∉ BR_k(x) then
5         Update strategy for player k to x_k ∈ BR_k(x);
6         L := ∅;
7     L := L ∪ {k};
```

In this program (BRD), L is the list of players that have played since the last change of the state $\mathbf{x}$, and it is reset to an empty list every time one player changes its action. As soon as this list reaches size n, the state $\mathbf{x}$ verifies the definition of a Nash equilibrium.

The worst case complexity of finding a Nash equilibrium in potential games is PLS complete [10], known to be between P and NP. In the following we focus on the expected execution time of BRD over a random potential game when the potential is chosen uniformly.

2.2. Randomization

In the following we will randomize over the potential games over which BRD is used. Since the behavior of BRD only depends on the potential function, we randomize directly over the potential Φ. On one hand, this is the classical average complexity approach when no additional information is known about the games (the same approach is used in [11] for 2 player games for example). This yields IID potential for all profiles, as explained below. On the other hand, some may argue that uniformly distributed random games are not generic in some sense and a good performance of BRD on average does not necessarily translate in good performances for "real world" games. In any case, we believe that in the absence of obvious structure in the potential games, uniform distribution over all potentials is a valid randomization.

There are several equivalent ways to do this randomization. The first one is based on the fact that the behavior of the algorithm does not depend on the actual values of the potential of the states but only on the comparisons between them. Therefore, the natural randomization is to consider all possible orderings of the state space $\mathcal{A}^n$ and pick one uniformly. The number of total orders on $\mathcal{A}^n$ is the number of permutations on $\mathcal{A}^n$, namely $(A^n)!$.

The second (equivalent but much more practical) randomization is the following: The potentials of all states x are chosen independent, identically distributed according to an arbitrary distribution F admitting a density w.r.t. the Lebesgue measure.

Both randomizations are equivalent. Indeed, take any k states $x_1, \ldots, x_k$ in $\mathcal{A}$. In both cases, $\mathbb{P}(\Phi(x_1) > \Phi(x_2) > \cdots > \Phi(x_k)) = 1/k!$. Now, since F is increasing over the support of its density, F^{-1} is well-defined and we get $\mathbb{P}(\Phi(x) > \Phi(x')) = \mathbb{P}(F^{-1}(\Phi(x)) > F^{-1}(\Phi(x')))$. Note that $F^{-1}(\Phi(x))$ is uniformly distributed on $[0, 1]$. Therefore, with no loss of generality, one can assume that the potential of all the states are i.i.d., uniformly distributed on $[0, 1]$. This randomization is used in the following.

Remark Randomizing over the payoffs of all players rather than over the potential (and rejecting those that do not satisfy the potential property) would provide a different randomization. However, this randomization is more difficult to analyze; it is not any more natural than randomizing the potential (especially if players are not homogeneous) and it does not substantially change the average behavior of BRD (this was checked numerically but it is not reported here,because this is out of the scope of this paper).

The average performance of Algorithm 1 under the round robin revision sequence for random games has been analyzed in [8]. This paper shows that the average execution time of Algorithm 1 to compute a Nash equilibrium is linear in the number of players. Furthermore, the time to reach a Nash equilibrium

on random games using BRD with a round robin revision sequence has been shown in [12] to be stochastically lower than with any other local search algorithm. In the distributed case, a central authority (or an election protocol among the players) is needed to enforce that the order of plays follows a round robin revision sequence. This can be either impossible or very costly. Therefore, it is interesting to investigate other playing sequences, and in particular revision sequences based on individual clocks for all players, that are more adapted to distributed games. This is precisely the focus of our paper.

3. Distributed Best Response

Let us now consider a distributed version of Algorithm 1 where players act according to individual Poisson clocks with rate λ/n. Apart from this change of the sequence of plays, the algorithm is the same as Algorithm 1: L is the list of players that are satisfied with the current state (they played under the current state and did not change their action).

Algorithm 2: Distributed BRD (for player k)

```
Input: Game utilities (u_k(·)); Initial state (x := x(0));
Initialize t := 0; List of satisfied customers L := ∅;
Local clock, ticking w.r.t. a Poisson process with rate λ/n;
while size(L)≠ n do
    On each tick of the local clock
        if k ∉ L then
            if x_k = BR_k(x) then
                L := L ∪ {k}
            else
                x_k := BR_k(x); L := {k}
```

In this distributed version of BRD, one can distinguish two phases. The first phase ends when the last change in the profile occurs (the last time when L is a singleton), at this point the algorithm has reached a Nash equilibrium but does not know it yet. The second phase is the time needed for all players to play and check that their best response in the current state dos not change, thus certifying the Nash equilibrium (this is the time needed for L to grow up to its maximal size, n). We will denote by R (*reaching time*) the duration of the first phase, and by T (*execution time*) the total time taken by the two phases. We denote by δ the time taken by one player (say k) to compute its best response, $\mathcal{BR}_k(\mathbf{x})$, under state $\mathbf{x}$. We assume that this time does not depend on the player nor on the current state, but may depend on $a = A - 1$ the number of alternative actions that a player needs to check before deciding its best response (for example, it could be linear in a). More precisely, if for one player, say k, the clock ticks at time t, then it takes δ amount of time for the

player to read the global state $\mathbf{x}$, compute its best response $\mathcal{BR}_k(\mathbf{x})$, and update its state to $x_k = \mathcal{BR}_k(\mathbf{x})$. We use the classical CREW-PRAM model for this distributed algorithm. The global variables are $\mathbf{x}$ and L, accessed by all players in a concurrent read and exclusive write mode. The PRAM model implies than the duration of play δ is constant over all times and all players, even if several plays overlap.

The main difference in the behavior of the distributed version of BRD vs the centralized one, is that the state may not be same for different players when their play overlap.

To make the point clear, let us consider a small example illustrated in Figure 1. The figure shows a time line of a game with three different players (say P_1, P_2, P_3), each with two possible actions and with clocks ticking respectively at times T_1, T_2, T_3, and their playing duration overlap as in Figure 1. Let us say that the initial state of the three players is $(0, 0, 0)$. Player P_1 plays first and changes its action for a better payoff to $x_1 = 1$. When player P_2 starts playing, P_1 has already finished its play, so P_2 compares the payoff of $(1, 0, 0)$ with $(1, 1, 0)$. Let us say that its best response is $x_2 = 1$. Meanwhile P_3 has started playing, because its play overlap with P_2. Its current state is still $(1, 0, 0)$, so P_3 compares the payoff of $(1, 0, 1)$ with $(1, 0, 0)$. Its best response is, say, $x_3 = 1$. This means that the state at time $T_3 + \delta$ will be $(1, 1, 1)$ whose potential has not been compared with the potential of $(1, 0, 0)$ (so it could be lower).

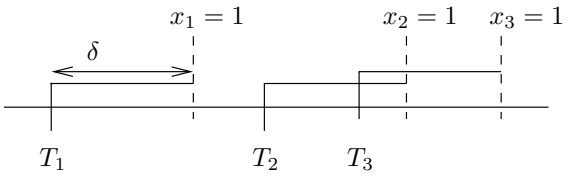

Figure 1: Time line of a distributed game when plays overlap

As shown in the previous example, overlaps can be seen as simultaneous plays and therefore the potential may decrease in this case. However convergence to a Nash equilibrium still holds with probability one: Using the Borel Cantelli Lemma, a sequence of round robin plays of arbitrary length and with no overlap will eventually occur, ensuring convergence. Of course it may take a very long time before such an event occurs.

Figure 2 shows the evolution of the potential during the execution of Algorithm 2 over one potential game with 50 players. As one can see, potential does not always increase, unlike what happens in the classical centralized case of Algorithm 1. However, convergence occurs eventually (after 60 steps in that case).

3.1. Main result

The rest of the paper is devoted to the proof that convergence occurs on average almost as fast with overlaps as when they do not happen (up to a multiplicative constant), as stated in the following theorem.

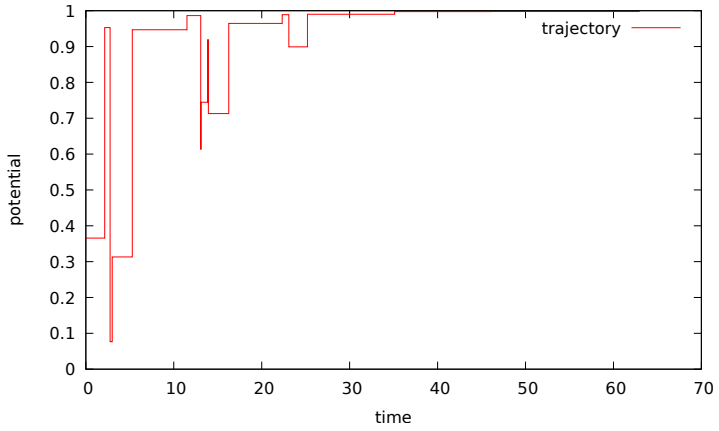

Figure 2: The evolution of the potential during the execution of Algorithm 2 up to the time when a Nash equilibrium is reached: Overlaps induce potential drops

Remark. In the following, most functions, such as $\mathbb{E}[T_{BRD}]$ or $G(p,n)$ defined below, will depend on $n \in \mathbb{N}$ (the number of players) and $p \in [0,1]$ (the overlap probability). To avoid tricky misleading terms in asymptotic developments (especially when p may depend on n), the notation $O(f(p,n))$ will have to be understood as a term asymptotically smaller than $f(n)$, uniformly in p. In other words, $h(p,n) \in O(f(p,n))$ if there exists a constant C such that

$$\lim_{n\to\infty} \sup_{p\in[0,1]} \frac{h(p,n)}{f(p,n)} \leq C.$$

The same convention applies to the notation $o(f(p,n))$, with a constant $C = 0$.

Theorem 1. *The expected execution time of the distributed best response dynamics with Poisson clocks of total rate λ as given in Algorithm 2 satisfies*

$$\frac{nH_n}{\lambda} \leq \mathbb{E}[T_{BRD}] \leq \frac{1}{\lambda}\left(n^{1-p}H_{n-1} + \frac{n^{1-p} + C_1 + \frac{C_2}{n} + C_2\frac{G(p,n)}{n(1-p)}}{1-p} + 1\right)G(p,n), \tag{1}$$

where $G(p,n) = \left(\frac{e^{pH_{n-1}}}{p} - \frac{1-p}{p}\right)$, H_n is the nth harmonic number, $p = 1-e^{-\delta\lambda}$ is the overlap probability, and C_1 and C_2 are constants.

The following bound can be given for $G(p,n)$: $\forall p \in [0,1]$, $\forall n \geq 2$, $G(p,n) \leq \frac{(e(n-1))^p}{p} - \frac{1-p}{p} \leq e(n-1)$. Indeed, the first inequality follows from $H_n \leq \log(n) + 1$, while the second is easily obtained by studying the function $f(x) := \frac{x^p}{p} - \frac{1-p}{p} - x$, noting that $f(1) = 0$ and f is non-increasing for all $x \geq 1$ (since its derivative is $f'(x) = x^{-(1-p)} - 1$), so that $f(x) \leq 0$ for all $x \geq 1$. This bound

on $G(p,n)$ yields a first simple bound for $\mathbb{E}[T_{BRD}]$, which is uniform in p:

$$\mathbb{E}[T_{BRD}] \leq \frac{e}{\lambda}\left(n^2\log(n) + \frac{n^2}{(1-p)^2}\right)(1+o(1)).$$

More refined bounds can be obtained, as follows. Let us denote by $B(n,p)$ the upper bound given in Theorem 1 (up to the factor $1/\lambda$), namely

$$B(n,p) := \left(n^{1-p}H_{n-1} + \frac{n^{1-p} + C_1 + \frac{C_2}{n} + C_2\frac{G(p,n)}{n(1-p)}}{1-p} + 1\right)G(p,n).$$

Its asymptotic behaviour, in different regimes, is the following:

- When p goes to 0, $B(n,p)$ goes to $e^\gamma nH_{n-1}^2 + O(nH_{n-1})$. This is a bound on the number of steps before convergence in the ideal case when $\delta = 0$. Actually, a tighter bound can be computed in this case: When $p = 0$, a direct proof (not reported here) shows that $\mathbb{E}[S_{BRD}] \leq e^\gamma nH_{n-1} + O(n)$.
- When p goes to 1, $B(n,p)$ goes to infinity, as expected: when all plays overlap, convergence in finite time cannot be guaranteed.
- When $p \in (0,1)$ is fixed and n goes to infinity, we have

  $$B(n,p) = \frac{e^\gamma}{p}\left(n\log(n) + \frac{n}{1-p}\right)(1+o(1)).$$

  This is obtained recalling that $\log(n-1)+\gamma+\frac{1}{2n} \leq H_{n-1} \leq \log(n-1)+\gamma+\frac{1}{2(n-1)}$, and hence $e^{pH_{n-1}} \leq (n-1)^p e^{p\gamma+\frac{p}{2(n-1)}}$.

Finally, letting p depend on n yields the following result.

Corollary 1. *Under a playing rate that minimizes the upper bound asymptotically, namely* $\hat{\lambda} = \frac{\log(\log(n))-\log\log\log(n))}{\delta}$, $\hat{p} = 1 - \frac{\log(\log(n))}{\log(n)}$, *the average execution time satisfies*

$$\frac{\delta n\log(n)}{\log(\log(n))}(1+o(1)) \leq \mathbb{E}[T_{BRD}] \leq e^\gamma\frac{\delta n\log(n)}{\log(\log(n))}(1+o(1)) \tag{2}$$

where γ is the Euler constant.

4. Markovian Approximation and Proof of Theorem 1

This section is devoted to the construction of a Markov model of BRD that leads to the proof of these bounds.

4.1. Intersection-Free Approximation (IFA)

The direct analysis of the behavior of BRD over a random potential is difficult because, over time, more and more states have been visited by the algorithm. Thus, its behavior is non-homogeneous in time. To avoid this difficulty, we consider a new model, called the Intersection Free Approximation (IFA) in the following. Under the Intersection Free Approximation, every time a new player (say k) who is not satisfied (in other words, a player not in L) has to compute its best response in a state (say $\mathbf{x}$), it compares $\Phi(\mathbf{x})$ with the potential of its a other strategies, as for the original BRD. However in IFA, differently from BRD, all the corresponding potentials will be randomly generated, whether or not those a states have been visited during the previous steps of the algorithm.

The IFA is given in Algorithm 3. The difference with Algorithm 2 is that every time a player plays, it generates the potential of its a alternative actions before computing its best response.

Algorithm 3: IFA Algorithm (for player k)

Input: Initial state ($\mathbf{x} := \mathbf{x}(0)$);
Local clock, ticking w.r.t. a Poisson process with rate λ/n;
List $L_{IFA} := \emptyset$;
while *size(L)*$\neq n$ **do**
  On *each tick of the local clock*
    if $k \notin L_{IFA}$ **then**
      forall $u \neq x_k$, **Generate** $\Phi(u, \mathbf{x}_{-k}) \sim$ Unif$[0, 1]$;
      if $x_k = \mathcal{BR}_k(\mathbf{x})$ **then**
        $L_{IFA} := L_{IFA} \cup \{k\}$
      else
        $x_k := \mathcal{BR}_k(\mathbf{x})$; $L_{IFA} := \{k\}$

To illustrate the difference between BRD and IFA, let us consider a simple two-player game with four actions per player, with a random potential. Such a potential is given in Figures 3 and 4. They respectively show the behavior of BRD and IFA under the revision sequence 1,2,1,2.

Under BRD, the potential of all states can be generated at the beginning, and the game is played under this fixed value of the potential (here uniform in $\{0, 1, \ldots, 10\}$).

Under IFA, potentials are generated each time a new player acts.

The first difference between the two systems occurs during the second play of player 2: one of the states that it can opt for has already been visited in the past in BRD. The potential of that state is known to be smaller than the current potential. Under IFA, the potential of this state is redrawn a second time, independently of the rest and can lead to a higher potential, as in Figure 4. Let V be the number of visited states whose utility (or potential in the case of IFA) has been evaluated during the execution.

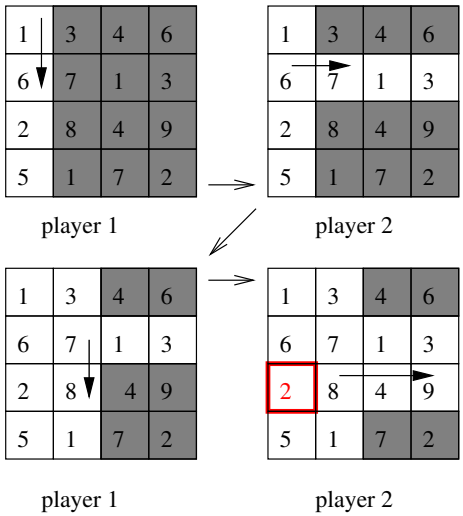

Figure 3: Evolution of the potential during BRD, under the revision sequence 1,2,1,2. The grey states have not been visited yet. When the second player plays for the second time, one state (in red) has already been visited.

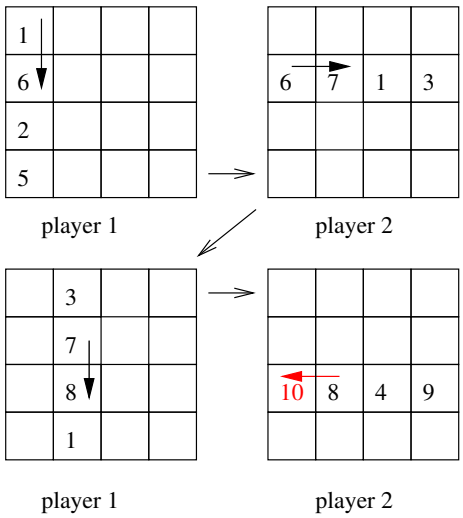

Figure 4: Evolution of the potential under IFA with the sequence 1,2,1,2. When the second player plays for the second time, the potential of state in red is redrawn and can become larger then the current potential.

Lemma 1 (Comparison with IFA).

(i) The number V_{BRD} of states visited by BRD is smaller than the number V_{IFA} of states visited by the IFA approximation, for the stochastic order.

(ii) The expected number of steps to reach convergence for best response ($\mathbb{E}S_{BRD}$) and IFA ($\mathbb{E}S_{IFA}$) verify:

$$\mathbb{E}S_{BRD} \leq \mathbb{E}S_{IFA} + \frac{C_1}{1-p}G(p,n) + C_2\frac{G(p,n)}{n(1-p)} + C_2\frac{G(p,n)^2}{n(1-p)^2},$$

where C_1 and C_2 are constants.

The proof is postponed to the Appendix.

4.2. Restart Approximation (RST)

Let us now consider the behavior of the IFA Algorithm. It should be clear that its future behavior only depends on the current potential $\Phi_{IFA}(t)$ and the current set $L_{IFA}(t)$, as long as there are no overlaps. When an overlap takes place, as seen in Figure 1, the current player will not consider the state computed by the previous player but the state obtained by the latest player with whom it is not overlapping. This makes the behavior of $X_{IFA}(t) \stackrel{\text{def}}{=} (\Phi_{IFA}(t), L_{IFA}(t))$ cumbersome to analyze because it is not Markovian in the potential.

We construct yet another approximation, of the behavior of IFA this time, whose state transition $X_{RST}(t) \stackrel{\text{def}}{=} (\Phi_{RST}(t), L_{RST}(t))$ is different from $X_{IFA}(t) = (\Phi_{IFA}(t), L_{IFA}(t))$ only when overlaps occur. This approximation will be called a *restart* version of IFA (denoted RST in the following). At any Poisson point (called t), if the current play overlaps with the previous one, who has been active (*i.e.* has changed its action), then, under the restart approximation, the state is reset to a uniform potential in $[0,1]$, disregarding the previously obtained potential and the set L is reset to a single player (the latest one).

More precisely, if $X_{RST}(t^-) = (\Phi, L)$ and the next player k' overlaps the previous play, then, $X_{RST}(t) = (U, \{k'\})$, where $U \sim \text{Unif}[0,1]$. In all other cases, restart behaves exactly as IFA. Note that by definition, $|L_{RST}(t)| = 1$ if $|L_{IFA}(t)| = 1$ for all t. Only the current potential differs when both sets are singletons.

The goal of the next Lemma is to compare the behavior of the two systems, and more precisely $\Phi_{IFA}(t)$ and $\Phi_{RST}(t)$.

Lemma 2. *For all t, $\Phi_{IFA}(t) \geq_{st} \Phi_{RST}(t)$.*

Proof. We compare IFA and RST by coupling the instants of plays of all players (denoted $(t_k)_{k\in\mathbb{N}}$), the order of play as well as the overlaps in both systems. One can refer to Figure 1 that displays the time line with one overlap.

To avoid cumbersome notations, we will keep the same notations for the original systems and the coupled systems. The proof holds by induction on t_k (the next playing instant). Let t_3 the first time that an overlap occurs while

the set of satisfied players is a singleton, as in Figure 1. Up to time t_2, both systems coincide under the previous coupling. We have $\Phi_{IFA}(t_1) = \Phi_{RST}(t_1)$. To deal with the general case directly, let us only use instead the weaker property $\Phi_{IFA}(t_1) \geq \Phi_{RST}(t_1)$, that will propagate by induction.

Under restart, the value of the potential U after any play in overlap (here, at time $t_3+\delta$) is uniformly distributed in $[0,1]$. In IFA, the player acting at time t_2 has changed its action. Therefore the potential $\Phi_{IFA}(t_2)$ must be higher than the potential $\Phi_{IFA}(t_1)$. By definition of IFA, this potential is the maximum of a uniformly distributed variables in the interval $[0,1]$, conditioned by the fact that it is larger than $\Phi_{IFA}(t_1)$. This implies that $\Phi_{IFA}(t_2) \geq_{st} U$, U being uniform in $[0,1]$.

Now let us consider the player acting at time t_3. It uses the state of the system at time t_2 to compute its best response. If the potential of its best response is not larger than $\Phi_{IFA}(t_2)$, which happens with positive probability, then this player does not change its action and the state at time $t_3 + \delta$ remains the state computed by the previous player: the potential is $\Phi_{IFA}(t_3) = \Phi_{IFA}(t_2) \geq_{st} U$. On the other hand, if the potential of its best response is larger than $\Phi_{IFA}(t_2)$, then it changes its action and the system reaches a new state where two players act simultaneously, whose potential under IFA $\Phi_{IFA}(t_0)$, is uniformly distributed in $[0,1]$. In total by considering both cases, $\Phi_{IFA}(t_0) \geq_{st} U$.

Now, if several overlaps occur in succession, the same type of reasoning concludes that with a positive probability (smaller and smaller as the number of overlaps increases) the potential remains the same as the potential of the first player, which is stochastically larger than a uniform one, and in the complementary case, the potential is newly generated with a uniform distribution.

In all cases, the potential remains stochastically larger than a uniform one, hence larger than the potential under RST. QED.

Corollary 2. *The time to reach a full set L is stochastically smaller under IFA than under the restart approximation.*

Proof. This result is a direct consequence of the previous lemma. Indeed, the order of play is the same for both systems as well as the occurrence of overlaps and since, at any time, the potentials are comparable, then if L_{IFA} is full at time t, L_{RST} must also be full at the same instant. Therefore, the first times of being full (implying termination in both systems) compare stochastically. QED
The construction of the restart approximation ensures that the future behavior of the system under RST only depends on its current state, namely the current potential and the list of satisfied players. It should also be clear that all states with the same number of satisfied players can be aggregated and the behavior remains Markovian, because the next player to play is IID, uniformly distributed among all players and does not depend on the previous plays. Therefore, in the following we focus on the evolution of the couple $(\Phi(t), |L(t)|)$ (potential, number of satisfied players).

4.3. Complexity of Restart

The sections above show that the average time for the original distributed best response algorithm with Poisson clocks to compute a Nash equilibrium is bounded from above by the time taken under the restart approximation.

The rest of this section is devoted to the computation of the average duration of RST.

Since all players play according to independent Poisson processes with rates λ/n, the global point process of active players $(T_i)_{i\in\mathbb{N}}$, forms a Poisson process with rate λ. In the following we focus on the behavior of the system at those points (called "instants" in the following).

It should be clear by now that $(\Phi(T_i), |L(T_i)|)$ is a discrete time Markov chain with a hybrid state space (a discrete component and a continuous one: the state space is $2^{\mathcal{N}} \times [0,1]$).

The average hitting time of the set of states $\{|L| = n)\}$ starting from a state with potential y and k satisfied players (including the current one) is given by a system of Poisson equations derived from a one-step analysis:

Let us denote by $s(y,k)$ the expected instant when a full set of satisfied players is reached (this corresponds to the termination of RST), starting with potential y and k satisfied players. One-step analysis yields the following equations, where $p = 1 - e^{-\lambda\delta}$ is the probability that the current play overlaps the previous one (since the playing times of all players form a Poisson process with rate λ).

$$\begin{aligned}
s(y,n) &= 0 \\
s(y,i) &= 1 + \frac{i}{n}s(y,i) + \frac{n-i}{n}y^a s(y,i+1) \\
&\quad + \frac{n-i}{n}\int_y^1 a u^{a-1} s(u,1)\mathrm{d}u, \quad n > i > 1 \\
s(y,1) &= 1 + \frac{n-1}{n}y^a(1-p)s(y,2) \\
&\quad + \frac{n-1}{n}(1-p)\int_y^1 a u^{a-1} s(u,1)\mathrm{d}u \\
&\quad + \frac{n-1}{n}p\int_0^1 s(u,1)\mathrm{d}u + \frac{1}{n}s(y,1)
\end{aligned}$$

These equations are derived using the behavior of restart. Starting in point $(y,i), i > 1$, the previous player did not modify its action so overlapping the current action with the previous one will not make any difference. This is why the equation for $s(y,i), i > 1$ does not depend on overlaps and hence does not involve p. Now, upon a play of a new player (this happens with probability $\frac{n-i}{n}$), the current profile will not change if this player does not find an action whose potential is above the current potential, namely y. This happens with probability $\frac{n-i}{n}y^a$. If the new player finds an action with potential $u > y$, then the current player becomes the only satisfied player and the potential goes to

u. Therefore, the Markov chain jumps to $(u,1)$. This happens with probability $\frac{n-i}{n} a u^{a-1} \mathrm{d}u$.

When the starting point is $(y,1)$, the situation is slightly more complex because overlaps do play a role.

Let us first consider the case with no overlap. This happens with probability $1-p$. In this case, the behavior is similar to the case $(y,i), i>1$. Now, when an overlap occurs, the potential is always reset to a uniform potential in $[0,1]$. This is precisely the effect of the restart approximation, and it happens with probability $\frac{n-1}{n}p$.

Defining $z := y^a$, $s^{(1)}(z,i) := s(z^{\frac{1}{a}},i)$, one gets

$$\begin{aligned}
s^{(1)}(z,n) &= 0;\\
s^{(1)}(z,i) &= \frac{n}{n-i} + \int_z^1 s^{(1)}(u,1)\,\mathrm{d}u + z s^{(1)}(z,i+1);\\
s^{(1)}(z,1) &= \frac{n}{n-1} + p\int_0^1 s(u,1)\,\mathrm{d}u\\
&\quad +(1-p)\int_z^1 s^{(1)}(u,1)\mathrm{d}u + (1-p) z\, s^{(1)}(z,2).
\end{aligned}$$

Rewriting the last equation as

$$\begin{aligned}
s^{(1)}(z,1) = p\frac{n}{n-1} + p\int_0^1 s(u,1)\,\mathrm{d}u\\
+ (1-p)\left[\frac{n}{n-1} + \int_z^1 s^{(1)}(u,1)\mathrm{d}u + z\, s^{(1)}(z,2)\right]
\end{aligned}$$

and telescoping all other equations into it, we obtain the following equation only involving $s^{(1)}(z,1)$.

$$\begin{aligned}
s^{(1)}(z,1) = \frac{pn}{n-1} + +p\int_0^1 s(u,1)\,\mathrm{d}u\\
+ (1-p)\sum_{j=0}^{n-2}\frac{n}{n-j-1} z^j + (1-p)\sum_{j=0}^{n-2} z^j \int_z^1 s^{(1)}(u,1)\,\mathrm{d}u. \quad (3)
\end{aligned}$$

We define $S(z) := \int_z^1 s^{(1)}(u,1)\mathrm{d}u$. Notice that $S(0)$ is the quantity we want to study. Also notice that $S'(z) = -s^{(1)}(z,1)$, so that (3) gives an ordinary differential equation for $S(z)$:

$$S'(z) + (1-p)\left[G(z)S(z) + V(z)\right] + pC_0 + D = 0,$$

where

$$G(z) := \sum_{i=0}^{n-2} z^i = \frac{1-z^{n-1}}{1-z}, \quad V(z) := \sum_{j=0}^{n-2}\frac{n z^j}{n-j-1},$$

$$C_0 := \int_0^1 s(u,1), \quad D := \frac{p\,n}{n-1},$$

We solve this equation on the interval $[0,1]$ using the boundary condition $S(1) = 0$. We obtain:

$$S(z) = -e^{-q(z)} \int_1^z \left((1-p)V(u) + pC_0 + D\right) e^{q(u)} \mathrm{d}u.$$

where $q(z) := (1-p)\int_0^z G(u)\,\mathrm{d}u = (1-p)\sum_{j=1}^{n-1} \frac{z^j}{j}$.

Since $q(0) = 0$, this gives

$$S(0) = (1-p)\int_0^1 V(u) e^{q(u)} \mathrm{d}u + (pC_0 + D)\int_0^1 e^{q(u)} \mathrm{d}u. \tag{4}$$

Notice that, for $z \in [0,1]$, $z^{\frac{1}{a}} \geq z$, and hence $s(z,1) \leq s^{(1)}(z,1)$, so that $C_0 = \int_0^1 s(z,1)\mathrm{d}u \leq \int_0^1 s^{(1)}(z,1)\mathrm{d}u = S(0)$. Replacing this bound in (4), we obtain

$$(1 - pQ_0)\, S(0) \leq (1-p)\int_0^1 V(u) e^{q(u)} \mathrm{d}u + \frac{n}{n-1} pQ_0, \tag{5}$$

with $Q_0 := \int_0^1 e^{q(u)} \mathrm{d}u$.

To study Q_0, we separate the integral in two terms. The first term is

$$\int_0^{1-b} e^{q(u)} \mathrm{d}u \leq \int_0^{1-b} e^{(1-p)\sum_{j=1}^{\infty} \frac{u^j}{j}} \mathrm{d}u = \int_0^{1-b} (1-u)^{p-1} \mathrm{d}u = \frac{1-b^p}{p}.$$

The second term is

$$\int_{1-b}^1 e^{q(u)} \mathrm{d}u \leq \int_{1-b}^1 e^{(1-p)\sum_{j=1}^{n-1} \frac{1}{j}} \mathrm{d}u = b e^{(1-p)H_{n-1}}.$$

Hence we have

$$Q_0 \leq \frac{1}{p} - \frac{b^p}{p} + b e^{(1-p)H_{n-1}}.$$

This bound is true for all $b \in (0,1)$. By using the best value for b, namely $b = e^{-H_{n-1}}$, we get

$$Q_0 \leq \frac{1}{p} - \frac{1-p}{p} e^{-pH_{n-1}} \tag{6}$$

and hence $1 - p\,Q_0 \geq (1-p)e^{-pH_{n-1}}$.

In particular, this shows that $1 - pQ_0 \geq 0$ and $pQ_0 \leq 1$, and hence from (5) we obtain

$$\begin{aligned} S(0) &\leq \frac{(1-p)\int_0^1 V(u) e^{q(u)} \mathrm{d}u + \frac{n}{n-1} pQ_0}{1 - pQ_0} \\ &\leq \frac{(1-p)\int_0^1 V(u) e^{q(u)} \mathrm{d}u + \frac{n}{n-1}}{1-p}\, e^{pH_{n-1}}. \end{aligned} \tag{7}$$

To study the numerator of (7), we define $Q_j := \int_0^1 u^j e^{q(u)} \mathrm{d}u$.
To bound Q_j let us first show that $Q_j \leq \frac{j}{j+p} Q_{j-1}$.
First consider $jQ_{j-1} = \int_0^1 j u^{j-1} e^{q(u)} \mathrm{d}u$. Integrating by parts, we have

$$\begin{aligned} jQ_{j-1} &= \left[u^j e^{q(u)}\right]_0^1 - \int_0^1 q'(u) u^j e^{q(u)} \mathrm{d}u \\ &= e^{(1-p)H_{n-1}} - (1-p) \sum_{i=0}^{n-2} \int_0^1 u^{j+i} e^{q(u)} \mathrm{d}u. \end{aligned}$$

Then consider $(j+p)Q_j = (j+1)Q_j - (1-p)Q_j$. For the term $(j+1)Q_j$ we do the same integration by parts as above, and hence we obtain

$$\begin{aligned} &(j+p)Q_j = (j+1)Q_j - (1-p)Q_j \\ &= \left[u^{j+1} e^{q(u)}\right]_0^1 - \int_0^1 u^{j+1} e^{q(u)} \mathrm{d}u - (1-p) \int_0^1 u^j e^{q(u)} \mathrm{d}u \\ &= e^{(1-p)H_{n-1}} - (1-p) \sum_{i=0}^{n-2} \int_0^1 u^{j+i+1} e^{q(u)} \mathrm{d}u - (1-p) \int_0^1 u^j e^{q(u)} \mathrm{d}u \\ &= e^{(1-p)H_{n-1}} - (1-p) \sum_{i=0}^{n-1} \int_0^1 u^{j+i} e^{q(u)} \mathrm{d}u \\ &= jQ_{j-1} - (1-p) \int_0^1 u^{j+(n-1)} e^{q(u)} \mathrm{d}u \end{aligned}$$

We have obtained $(j+p)Q_j \leq jQ_{j-1}$, which gives

$$Q_j \leq \frac{j}{j+p} Q_{j-1} \leq \left(\prod_{\ell=1}^{j} \frac{\ell}{\ell+p} \right) Q_0. \tag{8}$$

We introduce the following technical lemma that is useful to get an explicit bound for Q_j.

Lemma 3. *For all* $0 \leq p \leq 1$, $\prod_{\ell=1}^{j} \frac{\ell}{\ell+p} \leq (j+1)^{-p}$.

Proof. Let us define the function

$$F : p \mapsto \sum_{\ell=1}^{j} \log\left(\frac{\ell+p}{\ell}\right) - p \log(j+1).$$

Proving the lemma is equivalent to proving that $F(p) \geq 0$ for all $p \in [0,1]$. The second derivative of F is $F''(p) = \sum_{\ell=1}^{j} \frac{-1}{(\ell+p)^2}$, non-positive in $[0,1]$ implying that F is concave on $[0,1]$ and hence its minimum is either in 0 or in 1. $F(0) = 0$ and $F(1) = \sum_{\ell=1}^{j} \log\left(\frac{\ell+1}{\ell}\right) - \log(j+1) = -\log 1 + \log(j+1) - \log(j+1) = 0$, so F is non-negative on $[0,1]$. QED

Using Lemma (3), and $Q_j \leq \left(\prod_{\ell=1}^{j} \frac{\ell}{\ell+p}\right) Q_0$, we obtain

$$Q_j \leq (j+1)^{-p} Q_0$$

We are now ready to find a bound for the numerator in (7).

$$\begin{aligned}
(1-p)\sum_{j=0}^{n-2} \frac{nQ_j}{n-j-1} &\leq (1-p)Q_0 \sum_{j=0}^{n-2} \frac{n}{n-j-1}\frac{1}{(j+1)^p} \\
&= (1-p)Q_0 \sum_{j=1}^{n-1} \frac{(n-j)+j}{n-j}\frac{1}{j^p} \\
&= (1-p)Q_0 \left(\sum_{j=1}^{n-1} j^{-p} + \sum_{j=1}^{n-1} j^{1-p}\frac{1}{n-j}\right)
\end{aligned}$$

Finally, we notice that

$$\sum_{j=1}^{n-1} j^{-p} \leq \int_0^n x^{-p}\mathrm{d}x = \frac{n^{1-p}}{1-p}$$

and

$$\sum_{j=1}^{n-1} j^{1-p}\frac{1}{n-j} \leq (n-1)^{1-p}\sum_{j=1}^{n-1}\frac{1}{n-j} = (n-1)^{1-p}H_{n-1}.$$

Putting everything together we get

$$\begin{aligned}
S(0) &\leq \left(n^{1-p}H_{n-1} + \frac{n^{1-p}}{1-p} + \frac{n}{n-1}\right)\left(\frac{e^{pH_{n-1}}}{p} - \frac{1-p}{p}\right) \\
&\leq \left(n^{1-p}H_{n-1} + \frac{n^{1-p}}{1-p} + \frac{n}{n-1}\right) G(p,n), \quad (9)
\end{aligned}$$

where $G(p,n) = \left(\frac{e^{pH_{n-1}}}{p} - \frac{1-p}{p}\right)$.

4.4. Proof of Theorem 1

Equation (9) gives an upper bound for the average number of steps of a Nash equilibrium for the Restart approximation. Using Lemmas 1 and 2, we get an upper bound on the average number of steps taken by BRD to compute a Nash equilibrium.

Let S_{BRD} (resp. S_{IFA}, S_{RST}) be the number of steps in BRD (resp. IFA,

RST) before termination.

$$\begin{aligned}
\mathbb{E}[S_{BRD}] &\leq \mathbb{E}[S_{IFA}] + C_1\frac{G(p,n)}{1-p} + C_2\frac{G(p,n)}{n(1-p)} + C_2\frac{G(p,n)^2}{n(1-p)^2} \\
&\leq \mathbb{E}[S_{RST}] + C_1\frac{G(p,n)}{1-p} + C_2\frac{G(p,n)}{n(1-p)} + C_2\frac{G(p,n)^2}{n(1-p)^2} \\
&\leq \left(n^{1-p}H_{n-1} + \frac{n^{1-p} + C_1 + \frac{C_2}{n} + C_2\frac{G(p,n)}{n(1-p)}}{1-p} + 1 \right) G(p,n).
\end{aligned}$$

The expected time before getting to a Nash equilibrium is given by the formula $\mathbb{E}[T_{BRD}] = \frac{1}{\lambda}\mathbb{E}[S_{BRD}]$, that follows from Wald's lemma.

4.5. Rate Optimization: Proof of Corollary 1

One can choose the playing rate λ (or the overlap probability p) to minimize the expected time needed to obtain a Nash equilibrium. The overlap probability is $p = 1 - \exp(-\lambda\delta)$, so that $\lambda\delta = -\ln(1-p)$.

Since we only know a bound on $\mathbb{E}[T_{BRD}]$, we can instead compute the overlap probability $p^*(n)$ that minimizes that bound.

The optimal value can be estimated by differentiating and numerically solving $\frac{\partial \frac{-1}{\ln(1-p)} B(n,p)}{\partial p} = 0$. This was done using Maple. The behavior of $p^*(n)$ as n grows, is displayed in Figure 5. It shows that $p^*(n)$ is non-decreasing with n and grows from $p^*(3) \approx 0.65$ to $p^*(250) \approx 0.78$.

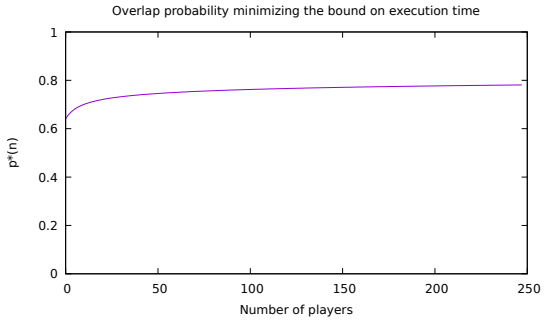

Figure 5: Overlap probability $p^*(n)$ minimizes the bound $\frac{1}{\lambda}B(n,p)$ on the expected convergence time $\mathbb{E}[T_{BRD}]$.

Analyzing the main term in the bound $B(n,p)$, it is direct to show that $p^*(n)$ goes to 1 as n goes to infinity.

Since $p^*(n)$ does not have a closed form, one can instead minimize the asymptotic behavior of $B(n,p)$. When p is close to 1 and n is large, the dominating terms are $n^{2p-1}/(1-p)^2$ and $n/(1-p)$ on one side and $n\log n$ on the other. This suggests choosing $\hat{p} = 1 - \log\log(n)/\log(n)$, with a corresponding playing rate

satisfying $\delta\hat{\lambda} = \log(\log(n)) - \log\log\log n$. The average execution time under $\hat{p}$ is now such that

$$\mathbb{E}[T_{BRD}] \leq e^{\gamma}\delta\frac{n\log(n)}{\log(\log(n))}(1+o(n)) \tag{10}$$

where $e^{\gamma} \approx 1.78$.

Finally, it should be clear that in BRD each player needs to play at least once to check if a state is a Nash equilibrium. The coupon collector theorem says that on average nH_n plays are necessary before all players have played once. Under rate $\hat{\lambda}$, each play takes $1/\hat{\lambda}$ units of time on average. This implies that

$$\mathbb{E}[T_{BRD}] \geq \delta\frac{n\log(n)}{\log(\log(n))}(1+o(n)).$$

This ends the proof of Theorem 1 and of its corollary.

5. BRD with Termination Test

We will now introduce a distributed algorithm using a convergence test that does not rely on the global variable L as before. A similar algorithm was presented in [13] in the case where players do not take any time to play their best response (this can be seen as the special case of our model with $\delta = 0$ and $p = 0$). In this algorithm, each player can still access the payoff of its actions but cannot know who is currently satisfied nor the global state of the system. To make termination possible, we let each player, when activated, also initiate a game-wide communication to make a termination test. Let us consider the communication procedures named *Termination Test Sender* and *Termination Test Receiver* in Algorithm 4: At every tick of its clock, each player has a probability q of broadcasting a message to every other player. Upon reception of such a message, the receivers interrupt their clock and send an acknowledgment (ack). Once the initial sender gets all the acks, it sends a second message. Upon reception of this second message, each player tests if it needs to change its best response (a player is *stable* if no change is needed) and sends back its stable/unstable status before restarting its clock. The initial sender receives n confirmations of stability only if the current state is a Nash equilibrium.

This global communication operation interrupts the Poisson clock of most players during two broadcasts and the clock of one (random) player for four broadcasts. We denote the average interruption time by σ_n.

Theorem 2. *When using the best value for q, the execution time of Algorithm 4, is*

$$\mathbb{E}[T^{(4)}] \quad \leq \quad \mathbb{E}[T_{BRD}] + 2\sqrt{\mathbb{E}[T_{BRD}]\sigma_n} + \sigma_n.$$

Algorithm 4: Distributed BRD with termination test

Function Main algorithm (for player k)
  Input: Game utilities ($u_k(\cdot)$); Initial state ($\mathbf{x} := \mathbf{x}(0)$);
  Local clock, ticking w.r.t. a Poisson process with rate λ/n;
  repeat
    On *each tick of the local clock*
      if $x_k \notin \mathcal{BR}_k(\mathbf{x})$ **then**
        Update strategy to $x_k \in \mathcal{BR}_k(\mathbf{x})$;
      With probability q:
       Call Termination Test Sender;
    On *Reception of* Stop
       Call Termination Test Receiver;
  until End *sent or received*;

Function Termination Test, Sender
  Stop Local clock;
  Send(Stop) to all players;
  wait until n acks received;
  Send(Test) to all players;
  wait until n messages received;
  if n 'Stable' messages received **then** **Send** End;
  Else Restart local clock;

Function Termination Test, Receiver (for player k)
  Stop Clock;
  Send(Ack) to p;
  wait until Test received;
  If $\mathcal{BR}(\mathbf{x}) = x_k$ **Send**(Stable) to sender;
  else **Send**(Unstable) to sender;
  Restart Clock;

Proof.

To study the execution time of Algorithm 4, one can separate the complexity of reaching the equilibrium with the complexity of the termination test. The second part is therefore independent of the overlaps and can be inspired from the approach used in the simpler case were overlaps are neglected, as in [13]. This execution time, denoted $T^{(4)}$, satisfies $T^{(4)} = W_1 + \cdots + W_k + k\sigma_n$, where W_i's are the times elapsed between two consecutive termination tests, and the random number k is the number of termination tests sent before a Nash equilibrium is reached. By construction of the algorithm, the random variables W_i's are independent and identically distributed, according to an exponential law of parameter $p\frac{\lambda}{n}$. Since W_i is independent of the event $\{k > i\}$, the expectation of $T^{(4)}$ can be computed with Wald's lemma:

$$\mathbb{E}[T^{(4)}] = \mathbb{E}[k]\mathbb{E}[W_1] + \mathbb{E}[k]\sigma_n. \tag{11}$$

On the other hand, this time is also the end of the first test after the reaching time of a Nash equilibrium, denoted $R^{(4)}$. The waiting time W from $R^{(4)}$ has the same exponential distribution as W_i's. Hence we also have $T^{(4)} = R^{(4)} + W + \sigma_n$.

Since the distributed algorithm uses Poisson clocks, the order of play is exactly as in Algorithm 2. Therefore, $\mathbb{E}[R^{(4)}] \leq (k-1)\sigma_n + \mathbb{E}[T_{BRD}]$. This yields

$$\mathbb{E}[T^{(4)}] = \mathbb{E}[T_{BRD}] + \mathbb{E}[W] + \mathbb{E}[k]\sigma_n. \tag{12}$$

By subtracting (12) from (11), and using $\mathbb{E}[W] = \mathbb{E}[W_1] = \frac{1}{q\lambda}$, one gets $\mathbb{E}[k] = q\lambda\mathbb{E}[T_{BRD}] + 1$.
Equation (12) becomes

$$\mathbb{E}[T^{(4)}] \leq \mathbb{E}[T_{BRD}] + \frac{1}{q\lambda} + q\lambda\mathbb{E}[T_{BRD}]\sigma_n + \sigma_n.$$

The best value for q that minimizes the formula above is $q^* = \frac{1}{\lambda\sqrt{\mathbb{E}[T_{BRD}]\sigma_n}}$.
Finally,

$$\mathbb{E}[T^{(4)}] \leq \mathbb{E}[T_{BRD}] + 2\sqrt{\mathbb{E}[T_{BRD}]\sigma_n} + \sigma_n.$$

QED

In particular, if one uses a classical model for global synchronization on a distributed algorithm (as in [14]), the duration of our two-steps broadcast is of the form $\sigma_n = O(\log(n))$. In this case, the execution time remains asymptotically bounded by the same value as for distributed BRD, namely

$$\mathbb{E}[T^{(4)}] \leq \delta C n \log(n)(1 + o(1)).$$

6. Sensitivity to the Playing Rate

In this part we simulate the behavior of the distributed versions of BRD presented in Algorithms 2 and 4 over a large number of potential games whose potentials are generated uniformly over all possible potential. These simulations

are carried to assess tightness of our bound and to study the sensitivity of BRD to the value of its unique parameter, namely λ.

The dependence on λ of the upper bound of the complexity of BRD given in Theorem 1 is explicit. However, the dependence on λ of the actual execution time remains unknown. We have run several experiments to evaluate the best value of λ numerically.

Figure 6 displays simulations of the distributed version of BRD as described in Algorithm 2 with 100 players, with 10 actions each (additional simulations show that the number of actions has no effect on the performance of BRD). For each value of λ, 1000 potential games were generated and the empirical mean reaching time of an equilibrium, R_{BRD}, have been computed, with 95% confidence intervals. The reaching time is hard to compute analytically because it is not a stopping time (it is the last time when that the state changes in BRD). However, it is a more interesting performance to measure than the total execution time T_{BRD} because the last phase (checking that the current state is an equilibrium) is completely equivalent to a coupon collector.

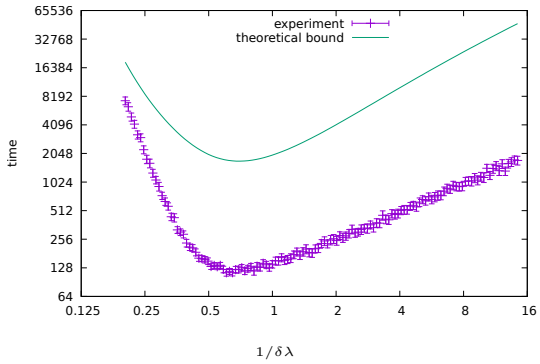

Figure 6: Mean reaching time of distributed BRD with 100 players (with 95% confidence intervals), as a function of the expected time between two plays (inverse of the playing rate), in log-log scale. Comparison with the analytic asymptotic bound

First, one can see that the mean reaching time is a relevant performance measure here: The confidence intervals are small. Furthermore, our numerous numerical simulations (not all reported here) never showed extreme outliers, where the execution time behaves as an exponential of the number of players (as suggested by the worst case analysis).

The main interest in Figure 6 is to compute the optimal value for λ. The formula in Section 4.5 says that $1/\delta\hat{\lambda} = 1.58$. On the other hand, the simulation measurements give $1/\delta\lambda_{opt} \approx 0.63$. The corresponding values for the overlap probabilities are $\hat{p} \approx 0.53$ for the analytic bound and $p_{opt} = 0.78$ from measurements. Both values are much larger than what our intuition would have predicted: overlaps are supposed to be bad for convergence. Actually this is not the case: the best situation is when more than two plays out of three are in overlap. The gap with the theoretical bound can be explained: In the restart

approximation, each overlap resets the potential to a uniform value in $[0, 1]$. which basically makes the games start all over again. In the actual behavior of BRD, some overlaps are harmless: even if an overlapping player uses an outdated value for the potential, this potential can still be larger than the potential of all its alternative actions. In this case the player becomes satisfied in spite of the overlap as if it were not overlapping with the previous play.

Figure 6 also shows (in log-log scale) a comparison with the bound given in Theorem 1. One can notice that the gap seems to remain constant in the log scale, suggesting a constant factor in λ between the two curves. The asymptotic slope is one in the log-log plot. This implies that the reaching time becomes linear in $1/\lambda$. This should not be surprising because as λ decreases, the probability of overlap also decreases so that the reaching time of the game becomes linear in the expected inter-playing time $(1/\lambda)$.

Finally, we have checked the dependence of the best rate λ_{opt} on n the number of players. The bound provided in Theorem 1 suggests the λ_{opt} should not depend on n. We have run simulations, all similar to those displayed in Figure 6, with several values of n (10, 30, 50, 70, 100, 130, 160, 200, 250). In each case, the optimal value $\lambda_{opt}(n)$ was evaluated numerically (the code and all simulation data are available upon request).

In a nutshell, our experiments show that λ_{opt} does not depend on n: For any n, $1/\delta\lambda_{opt} \approx 0.63$ implying $p_{opt} \approx 0.77$. However it was very hard to get a good precision on λ_{opt}. Even if these experiments took several days of computations over a modern desktop computer, with 1000 runs for each couple (n, λ), our accuracy on the optimal value of $\lambda_{opt}(n)$ is between 5% and 10%, depending on the value of n.

On the other hand, the best empirical mean reaching time R_{opt} was easier to evaluate numerically in these simulations and it is reported in Figure 7. A least square regression suggests that $R_{opt} = C_5 n^p$ for some constant C_5, and p coincides almost perfectly with p_{opt}, the probability of overlap under the optimal rate λ_{opt}. Under this numerical model, the total execution time is $\mathbb{E}[T_{opt}] = C_5 \delta n^p + \frac{1}{\lambda} n H_n$.

7. Conclusion

This paper presents an analytic study of the complexity of Best Response Dynamics to compute Nash equilibria of potential games in a distributed setting. On a methodological point of view, it shows the accuracy and the power of the Markovian approach to analyze algorithms: even if our analysis goes through several upper bounds at the model level (IFA and RST) as well as in the calculations, it only introduces a constant factor in the complexity. Simulations suggest that the actual complexity with the best playing rate λ_{opt} is $1/\lambda_{opt}(nH_n + C_5 n^p)$, where p is the probability of overlap under the optimal rate λ_{opt}.

This result also suggests that in spite of its simplicity and greediness, BRD is very efficient to compute Nash equilibria in potential games in a distributed

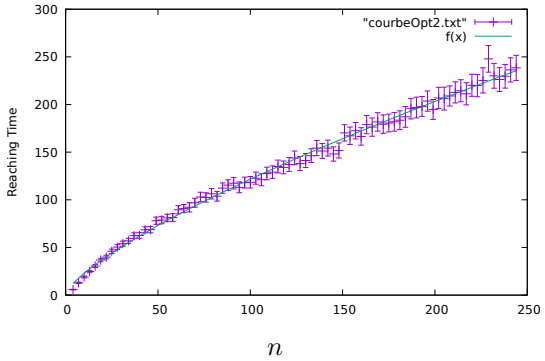

Figure 7: Average reaching time as a function of n (with 95% confidence intervals) and the best fit of the form $f(n) = Cn^p$, when the rate is λ_{opt}.

context where no synchronization of the players is possible. Two factors could hamper the performance of BRD in such a setting: random order of plays and overlaps. We show that a random order of plays only introduces a $\log(n)$ factor (due to a coupon collector effect) while overlaps introduce a factor $1/(1-p)$ (that can be controlled by tuning the rate λ). We claim that BRD should be considered as a serious option to solve distributed games when no additional structure except that of having a potential is available on the game.

Acknowledgement

This work was supported by the LabEx PERSYVAL-Lab.

Appendix A. Proof of Lemma 1

The main idea of the proof is to construct a coupling of BRD and IFA under which they can be compared. To do that, let us consider an infinite sequence $U = (U(1), U(2), \ldots)$ of IID random variables, uniformly distributed in $[0, 1]$. We couple an execution of the original BRD with an execution of IFA as follows. In both executions, each new potential used (or visited) by the algorithm is the next value in the sequence U. This coupling is different from the natural coupling used in Figures 3 and 4.

Appendix A.1. Number of visited states

Under this coupling, we will show an almost sure comparison, $V_{BRD} \leq V_{IFA}$.

Let us consider that the tth state visited by IFA is its final state. This implies that all the values of the potential Φ in the sequence $(U(t+1), U(t+2), \ldots, U(t+(A-1)N)$ are all below $\Phi(U(t))$.

Let us assume that after $t-1$ comparisons with other potentials, BRD has not stopped yet. Then $U(t)$ is used while the current player computes its best

response. There are at most $A-2$ other states that will be visited by the current player. The state obtained after this player has finished computing its best response is a Nash equilibrium if all remaining unsatisfied players agree. There are at most $n-1$ of them. In total they will explore less that $(A-1)(n-1)$ values in the sequence U. We already know that all these states have a potential below $\Phi(U(t))$. This implies that the state with potential $U(t)$ will also be a stopping state for BRD. This implies that $V_{BRD} \leq V_{IFA}$.

Appendix A.2. Number of Steps

Let us consider the number of plays (or setps), denoted S. The number of potentials visited per step under IFA is exactly $A-1$ while BRD visits $A-1-I_t$ new potentials at each step t, where I_t is the number of *intersections* for the current step t, *i.e.* the number of states visited by the current player that have already been visited in the past. So in total,

$$S_{BRD} = \frac{V_{BRD}+I}{A-1} \leq \frac{V_{IFA}+I}{A-1} = S_{IFA} + \frac{I}{A-1}, \tag{A.1}$$

where $I \stackrel{\text{def}}{=} I_1 + \cdots + I_{T_{BRD}}$ is the total number of intersections.

Appendix A.3. Number of Intersections

This section is devoted to studying the number of intersections. We start by decomposing it as a sum over the number M of moves (number of state changes during the execution of the algorithm):

$$I = \sum_{i=0}^{M-1} I(i),$$

where $I(i)$ is the number of intersections of the current state $\mathbf{x}(i)$ (reached after i moves) with all future states, visited at moves $i+1, i+2, \ldots, M$.

Let $d(\mathbf{x}, \mathbf{y})$ be the Hamming distance between states $\mathbf{x}$ and $\mathbf{y}$, namely the number of different coordinates.

Notice that, if the distance $d(\mathbf{x}_i, \mathbf{x}_j)$ is 2, then two intersections are possible with plays in states $\mathbf{x}_j$ and state $\mathbf{x}_j$. When the distance is not 2, no intersection can take place.

By conditioning on the value of $M = m$ (the conditional expectation is denoted $\mathbb{E}^{(m)}$),

$$\mathbb{E}^{(m)} I = \mathbb{E}^{(m)} \sum_{i=0}^{m-1} I(i) = \sum_{i=0}^{m-1} \mathbb{E}^{(m)} I(i) \leq \sum_{i=0}^{m-1} 2\,\mathbb{E}^{(m)} \sum_{j=i+1}^{m-1} \mathbf{1}_{(d(\mathbf{x}_i,\mathbf{x}_j)=2)}. \tag{A.2}$$

We study $H(k,i) := \sum_{j=i+1}^{m-1} \mathbf{1}_{(d(\mathbf{x}_i,\mathbf{x}_j)=2)}$, namely the number of times the Hamming distance is 2. The evolution of the Hamming distance depends the

moves. Let us denote by k_t the player that was active at the t-th move. The Hamming distance evolution over the moves depends on the current Hamming distance and on whether the k_t-th coordinate of $\mathbf{x}_i$ is equal or not to the k_t-th coordinate of the current state.

We construct a discrete-time Markov chain X_t with four states A, B, C, and D, where state A represents distance 1, states B represents distance 2 (with different coordinates), C represent distance 2 with equal coordinates and state D represents distance 3 or more. We choose transition probabilities such that the number of visits to states B and C will be an upper bound on $H(2,i)$.

This chain has the following transition matrix:

$$P = \frac{1}{n-1}\begin{pmatrix} 0 & n-1 & 0 & 0 \\ \frac{1}{a} & 1-\frac{1}{a} & 0 & n-2 \\ \frac{2}{a} & 2-\frac{2}{a} & 0 & n-3 \\ 0 & 0 & \frac{3}{a} & n-1-\frac{1}{a} \end{pmatrix},$$

An upper bound for $H(2,i)$ is obtained by counting the number of visits in states B and C in $m-i$ time steps. As a further upper bound, we have

$$\mathbb{E}^{(m)}H(2,i) \leq \mathbb{E}_{\mathrm{B}}(\#\text{ visits to B before } S_{\mathrm{B}}) + \mathbb{E}_{\mathrm{C}}(\#\text{ visits to C before } S_{\mathrm{C}}), \tag{A.3}$$

where $\mathbb{E}_x(\cdot)$ denotes the conditional probability $\mathbb{E}(\cdot\,|X_0 = x)$ and S_x is a stopping time defined by $S_x := \min\{t \geq m-i \text{ s.t. } X_t = x\}$. Notice that X_t is an irreducible Markov chain, and S_x is a stopping time such that $\mathbb{E}_x(S_x) < \infty$ and $X_{S_x} = x$. Hence, by Prop. 3 in [15, Chapter 2],

$$\mathbb{E}_x(\#\text{ visits to } x \text{ before } S_x) \leq \pi_x \mathbb{E}_x(S_x), \tag{A.4}$$

where π is the stationary distribution of the Markov chain. Then notice that

$$\mathbb{E}_x(S_x) = m - i + \sum_y [P^{m-i}]_{xy}\mathbb{E}_y(T_x), \tag{A.5}$$

where T_x is the first hitting time of state x. By Lemma 12 in [15, Chapter 2],

$$\pi_x \mathbb{E}_y(T_x) = Z_{xx} - Z_{yx},$$

where $Z := \sum_{j=0}^{\infty}(P^j - \mathbf{1}\pi^T)$ is the fundamental matrix of the Markov chain (here $\mathbf{1}$ denotes a column vector with four ones). Hence, we obtain

$$\pi_x \sum_y [P^{m-i}]_{xy}\mathbb{E}_y(T_x) = \sum_x [P^{m-i}]_{xy}(Z_{xx} - Z_{yx}) = [(I - P^{m-i})Z]_{xx}. \tag{A.6}$$

Using the definition of Z and the fact that $\lim_{j\to\infty} P^j = \mathbf{1}\pi^T$ (since the Markov chain is irreducible and aperiodic), we obtain

$$(I - P^{m-i})Z = \sum_{j=0}^{\infty}(P^j - P^{j+m-i}) = \sum_{j=0}^{m-i-1}(P^j - \mathbf{1}\pi^T) = I - \mathbf{1}\pi^T + \sum_{j=1}^{m-i-1}(P - \mathbf{1}\pi^T)^j$$

and hence

$$[(I-P^{m-i})Z]_{xx} = 1-\pi_x + \sum_{j=1}^{m-i-1} [P-\mathbf{1}\pi^T)^j]_{xx} \leq -\pi_x + \sum_{j=0}^{\infty} \|(P-\mathbf{1}\pi^T)^j\|_{\max}, \tag{A.7}$$

where $\|\cdot\|_{\max}$ denotes the maximum modulus of all entries in a matrix.

Using equations (A.4)-(A.7) with $x = \mathrm{B}$ and $x = \mathrm{C}$, from the bound (A.3) we obtain

$$\mathbb{E}^{(m)}H(2,i) \leq (\pi_{\mathrm{B}} + \pi_{\mathrm{C}})(m-i-1) + 2\sum_{j=0}^{\infty} \|(P-\mathbf{1}\pi^T)^j\|_{\max}\,. \tag{A.8}$$

Explicit computation of π, $(P-\mathbf{1}\pi^T)$ and $(P-\mathbf{1}\pi^T)^2$ shows that $\pi_{\mathrm{B}} + \pi_{\mathrm{C}} = \frac{3}{a(n-3)}$, and that, for some positive constants $c_1 \leq 2$ and c_2, $\|(P-\mathbf{1}\pi^T)\|_{\max} \leq c_1$ and $\|(P-\mathbf{1}\pi^T)^2\|_{\max} \leq c_2/n$. This implies that, for all $j \geq 1$, $\|(P-\mathbf{1}\pi^T)^j\|_{\max} \leq c_1(4c_2/n)^{(j-1)/2}$ if j is odd, and $\|(P-\mathbf{1}\pi^T)^j\|_{\max} \leq 4^{\frac{j}{2}-1}(c_2/n)^{j/2}$ if j is even. Hence,

$$\sum_{j=0}^{\infty} \|(P-\mathbf{1}\pi^T)^j\|_{\max} \leq 1 + \sum_{h=0}^{\infty} \left(c_1 + \frac{1}{4}\right)\left(\frac{4c_2}{n}\right)^h = 1 + \left(c_1 + \frac{1}{4}\right)\frac{1}{1-4c_2/n},$$

where the last equality is true for $n > 4c_2$. Plugging these bounds into (A.8), we obtain

$$\mathbb{E}^{(m)}H(2,i) \leq \frac{3}{a(n-3)}(m-i-1) + 2 + \left(2c_1 + \frac{1}{2}\right)\frac{n}{n-4c_2}.$$

Recalling that (A.2) gives $\mathbb{E}^{(m)}I \leq 2\sum_{i=0}^{m-1} \mathbb{E}^{(m)}H(2,i)$, we have

$$\mathbb{E}^{(m)}I \leq \frac{6}{a(n-3)} \sum_{i=0}^{m-1}(m-i-1) + \left(4 + (4c_1+1)\frac{n}{n-4c_2}\right)m.$$

Using $\sum_{i=0}^{m-1}(m-i-1) = (m-1)m/2 \leq m^2/2$ and deconditioning, we finally obtain

$$\mathbb{E}I \leq \frac{3}{a(n-3)}\mathbb{E}(M^2) + \left(4 + (4c_1+1)\frac{n}{n-4c_2}\right)\mathbb{E}(M). \tag{A.9}$$

Appendix A.4. Expected Number of Moves

The analysis of the expected number of moves, follows the same path as for the execution time.

First, one needs to show that the number of moves is larger under the IFA approximation. This is done by using the same coupling as for the number of steps. Then, one also needs to show that the number of moves is larger under the Restart approximation. Again, the proof is done by coupling IFA and Restart. By the same coupling argument as in the proof of $V_{BRD} \leq_{st} V_{IFA}$, we can show

that $M_{BRD} \leq_{st} M_{IFA} \leq_{st} M_{RST}$. Finally, it is also easy to see that the case $A = 2$ is an upper bound on the case where $A \geq 2$.

The expected number of moves for Restart satisfies a Poisson equation similar to the equation for the number of steps:

Let us denote by $m(y, k)$ the average number of moves before reaching a Nash equilibrium, starting with potential y and k satisfied players. One-step analysis yields:

$$\begin{aligned}
m(y,n) &= 0 \\
m(y,i) &= \frac{i}{n}m(y,i) + \frac{n-i}{n}ym(y,i+1) \\
&\quad + \frac{n-i}{n}\int_y^1 (m(u,1)+1)\mathrm{d}u, \quad n > i > 1; \\
m(y,1) &= \frac{1}{n}m(y,1) + \frac{n-1}{n}(1-p)ym(y,2)) \\
&\quad + \frac{n-1}{n}(1-p)\int_y^1 (m(u,1)+1)\mathrm{d}u \\
&\quad + \frac{n-1}{n}p\int_0^1 m(u,1)\mathrm{d}u.
\end{aligned}$$

By simplifying, we get

$$\begin{aligned}
m(y,n) &= 0 \\
m(y,i) &= 1 - y + ym(y,i+1) \\
&\quad + \int_y^1 m(u,1)\mathrm{d}u, \quad n > i > 1; \\
m(y,1) &= 1 - y + yp - p + y(1-p)m(y,2) \\
&\quad + (1-p)\int_y^1 m(u,1)\mathrm{d}u \\
&\quad + p\int_0^1 (m(u,1)+1)\mathrm{d}u.
\end{aligned}$$

We define $M(z) := \int_z^1 m(u,1)\mathrm{d}u$ and use telescoping so that the previous equations become a single ordinary differential equation with implicit initial condition $M(0)$:

$$m(y,1) = (1-p)\sum_{j-0}^{n-2} y^i M(y) + (1-p)\sum_{j-0}^{n-2} y^i(1-y) + yp + pM(0).$$

This equation admits an explicit solution for $M(0)$, using the same function

$q(u)$ as before.

$$\begin{aligned} M(0) &= \frac{\int_0^1((1-p)(1-u^{n-1})+pu)e^{q(u)}\mathrm{d}u}{1-p\int_0^1 e^{q(u)}\mathrm{d}u} \\ &\leq \frac{(1-p)Q_0+pQ_1}{1-pQ_0} \\ &\leq \frac{Q_0}{1-pQ_0}. \end{aligned}$$

Using the bound $Q_0 \leq \frac{1}{p} - \frac{1-p}{p}e^{-pH_{n-1}}$, given in (6),

$$M(0) \leq \frac{1}{1-p}\left(\frac{e^{pH_{n-1}}}{p} - \frac{1-p}{p}\right) = \frac{1}{1-p}\,G(p,n). \tag{A.10}$$

As for the second moment, let $m_2(y,1) = \mathbb{E}(M(y,k)^2)$ denote the second moment of the number of moves, starting with potential y and k satisfied players. One step analysis yields the following Poisson equation, involving both m_2 and m.

$$\begin{aligned} m_2(y,n) &= 0 \\ m_2(y,i) &= ym_2(y,i+1) \\ &\quad + \int_y^1 (m_2(u,1)+2m(u,1)+1)\mathrm{d}u, \quad n > i > 1 \\ m_2(y,1) &= y(1-p)m_2(y,2) \\ &\quad +(1-p)\int_y^1 (m_2(u,1)+2m(u,1)+1)\mathrm{d}u \\ &\quad +p\int_0^1 (m_2(u,1)+2m(u,1)+1)\mathrm{d}u \end{aligned}$$

By telescoping all $m_2(y,i), i>1$ and using $M_2(z) := \int_z^1 m_2(u,1)\mathrm{d}u$,

$$\begin{aligned} m_2(y,1) &= (1-p)\sum_{j=0}^{n-2} z^j M_2(z) \\ &\quad +(1-p)\sum_{j=0}^{n-2} z^j((1-z)+2M(z)) \\ &\quad +p+2pM(0)+pM_2(0). \end{aligned}$$

This yields an integral form for $M_2(0)$

$$M_2(0 = \frac{\int_0^1 e^{q(z)}(p+2pM(0)+(1-p)(1-z^{n-1})+2(1-p)\sum_{j=0}^{n-2} z^j M(z))\mathrm{d}z}{1-p\int_0^1 e^{q(u)}\mathrm{d}u}$$

Using the fact that $M(z) = \int_z^1 m(u,1)\mathrm{d}u$, and $m(u,1)$ is decreasing, we get $M(z) \leq (1-z)\int_0^1 m(u,1)\mathrm{d}u = (1-z)M(0)$. This implies that

$$\begin{aligned} M_2(0) &\leq \frac{Q_0 + 2Q_0M(0)}{1-pQ_0} \\ &\leq \frac{Q_0}{1-pQ_0} + 2\left(\frac{Q_0}{1-pQ_0}\right)^2 \\ &\leq \frac{1}{1-p}G(p,n) + \frac{2}{(1-p)^2}G(p,n)^2 \end{aligned}$$

where the second inequality uses (A.10).

In conclusion,

$$\mathbb{E}(M) \leq \frac{1}{1-p}G(p,n), \tag{A.11}$$

$$\mathbb{E}(M^2) \leq \frac{1}{1-p}G(p,n) + \frac{2}{(1-p)^2}G(p,n)^2. \tag{A.12}$$

Appendix A.5. End of the Proof

The proof of Lemma (1) follows directly from Equations (A.9) and (A.1) together with (A.11) and (A.12).

References

[1] R. W. Rosenthal, A class of games possessing pure-strategy Nash equilibria, Int. J. of Game Theory, Springer 2 (1) (1973) 65–67.

[2] M. Beckman, C. B. McGuire, C. B.Winsten, Studies in the Economics of Transportation, Yale University Press, 1956.

[3] D. Monderer, L. Shapley, Potential games, Games and economic behavior, Elsevier 14 (1) (1996) 124–143.

[4] A. Orda, R. Rom, N. Shimkin, Competitive routing in multi-user communication networks, IEEE/ACM Trans. on Networking 1 (5) (1993) 510–521.

[5] R. G. Gallager, A minimum delay routing algorithm using distributed computation, IEEE Transactions on Communications 25 (1) (1977) 73–85.

[6] J. Wardrop, Some theoretical aspects of road traffic research. Part ii, Proc. of the Institute of Civil Engineers 1 (1954) 325–378.

[7] T. Roughgarden, Selfish Routing and the Price of Anarchy, MIT Press, 2005.

[8] S. Durand, B. Gaujal, Complexity and optimality of the best response algorithm in random potential games, in: Symposium on Algorithmic Game Theory (SAGT) 2016, Liverpool, United Kingdom, 2016, pp. 40–51. doi:10.1007/978-3-662-53354-3_4.

[9] M. Voorneveld, Best-response potential games, Economics letters 66 (3) (2000) 289–295.

[10] A. Fabrikant, C. Papadimitriou, K. Talwar, The complexity of pure Nash equilibria, in: Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing, STOC '04, ACM, 2004, pp. 604–612.

[11] B. Imre, S. Vempala, A. Vetta, Nash equilibria in random games, Random Structures and Algorithms 31 (4) (2007) 391–405.

[12] S. Durand, B. Gaujal, Complexity and optimality of the best response algorithm in random potential games, Research Report RR-8925, Inria (Jun. 2016).
URL `https://hal.inria.fr/hal-01330805`

[13] S. Durand, B. Gaujal, F. Garin, Distributed best response algorithms for random potential games, in: European Control Conference, 2018.

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, Introduction to Algorithms, Third Edition, MIT Press, 2009.

[15] D. Aldous, J. A. Fill, Reversible Markov chains and random walks on graphs, unfinished monograph, recompiled 2014, available at `www.stat.berkeley.edu/$\sim$aldous/RWG/book.html` (2002).